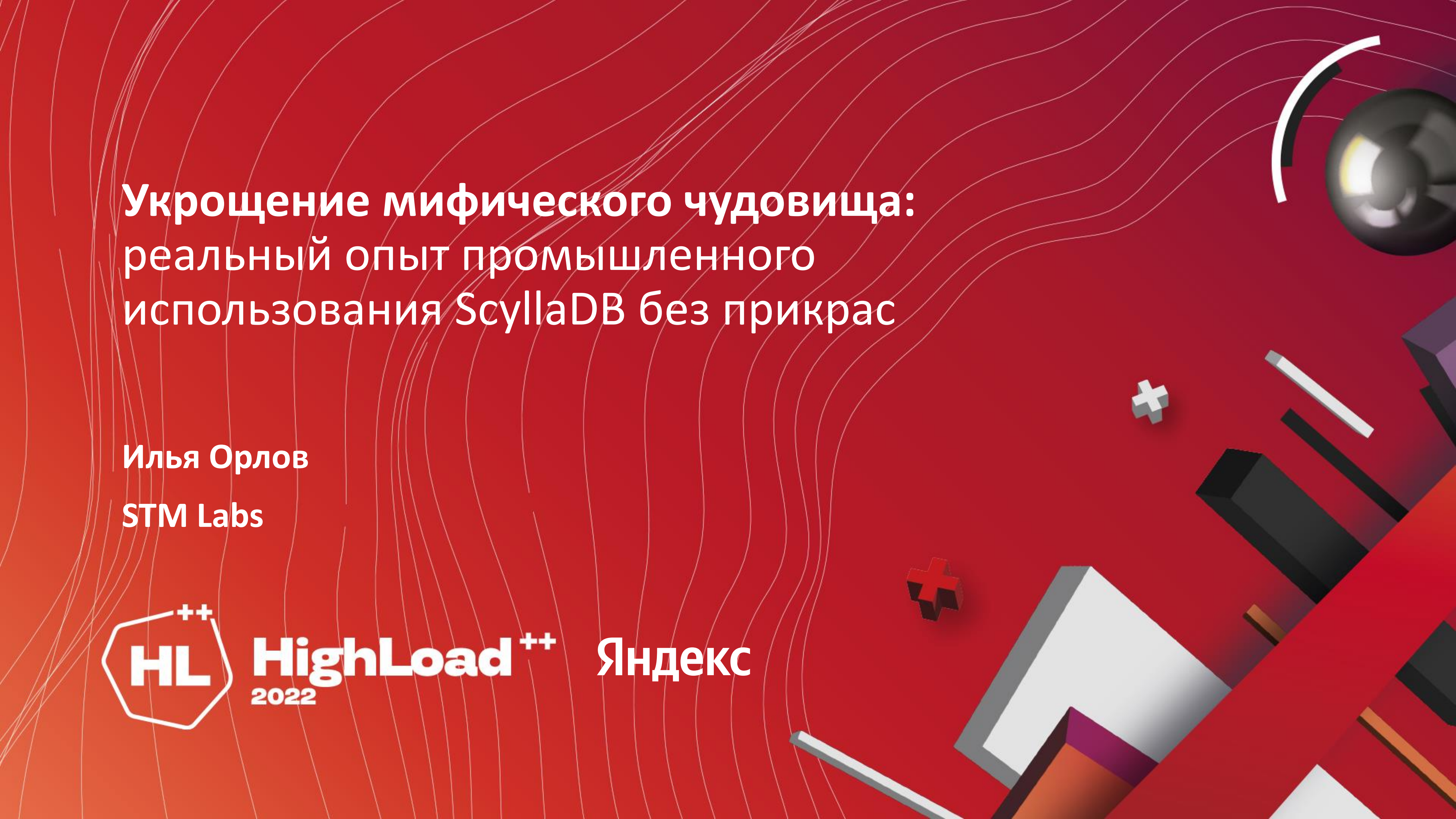

Укрощение мифического чудовища:
реальный опыт промышленного
использования ScyllaDB без прикрас
Илья Орлов
STM Labs
HL
HighLoad++
2022
Яндекс



УКРОЩЕНИЕ МИФИЧЕСКОГО ЧУДОВИЩА:

реальный опыт промышленного использования ScyllaDB без прикрас

Илья Орлов

www.stm-labs.ru

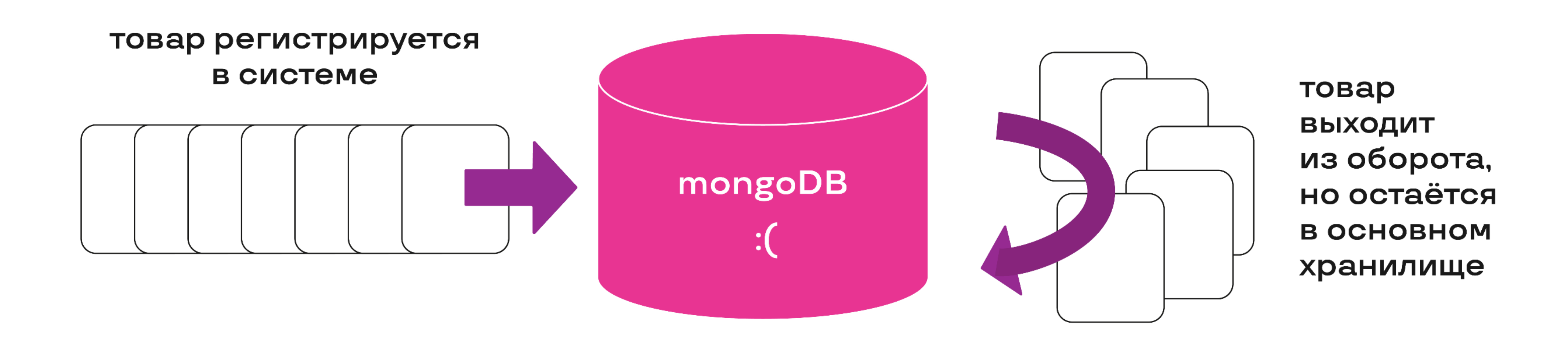
Oh....really?

Яндекс HighLoad++ 2022

Мы любим эксперименты

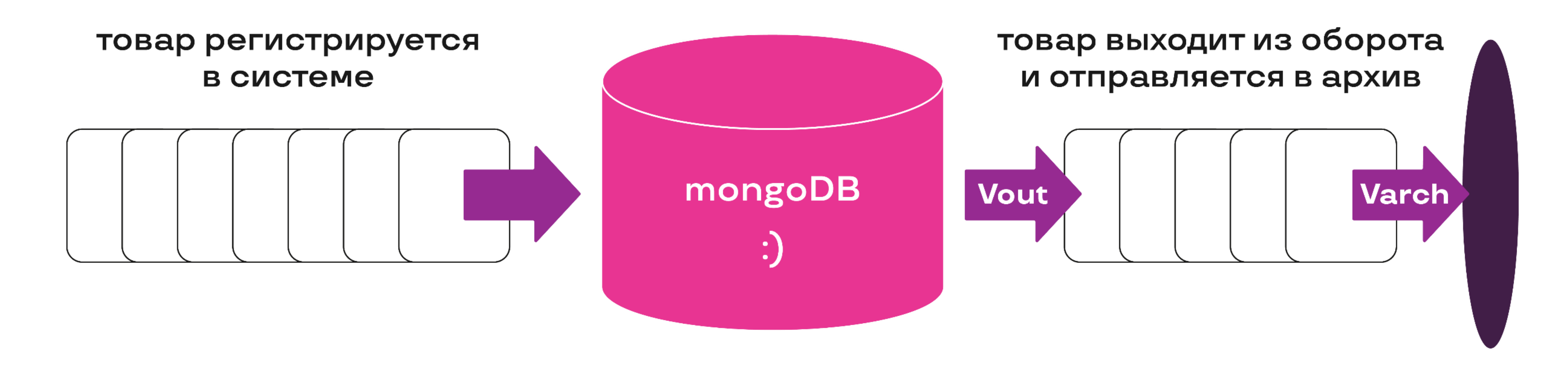


Когда решил почистить базу,
От данных диск освободив,
Не торопись стирать их сразу,
Пускай сперва идут...





В архив!



Требуемая скорость архивации: **Varch** >= **Vout** (30 млн записей в сутки)

Средний размер записи: 10 Кбайт

Varch >= 3,4 Мбайта/с



О, Сцилла!

1. Скорость записи — 12500 rps/core
   (~400 Мбайт/с на 32-ядерном сервере)



О, Сцилла!

1. Скорость записи — 12500 rps/core
   (~400 Мбайт/с на 32-ядерном сервере)
2. Реализация на C++



О, Сцилла!

1. Скорость записи — 12500 rps/core
   (~400 Мбайт/с на 32-ядерном сервере)
2. Реализация на C++
3. Восторженные отзывы (с официального сайта)



О, Сцилла!

1. Скорость записи — 12500 rps/core
 (~400 Мбайт/с на 32-ядерном сервере)
2. Реализация на C++
3. Восторженные отзывы (с официального сайта)
4. Эконом-вариант (HDD): 240000 rps (~240 Мбайт/с)



О, Сцилла!

1. Скорость записи — 12500 rps/core (~400 Мбайт/с на 32-ядерном сервере)
2. Реализация на C++
3. Восторженные отзывы (с официального сайта)
4. Эконом-вариант (HDD): 240000 rps (~240 Мбайт/с)
5. Привычный интерфейс (как у CassandraDB)



О, Сцилла!

1. Скорость записи — 12500 rps/core
   (~400 Мбайт/с на 32-ядерном сервере)
2. Реализация на C++
3. Восторженные отзывы (с официального сайта)
4. Эконом-вариант (HDD): 240000 rps (~240 Мбайт/с)
5. Привычный интерфейс (как у CassandraDB)
6. Море доступной документации



О, Сцилла!

1. Скорость записи — 12500 rps/core (~400 Мбайт/с на 32-ядерном сервере)
2. Реализация на C++
3. Восторженные отзывы (с официального сайта)
4. Эконом-вариант (HDD): 240000 rps (~240 Мбайт/с)
5. Привычный интерфейс (как у CassandraDB)
6. Море доступной документации
7. Открытое сообщество на Github и поддержка в Slack



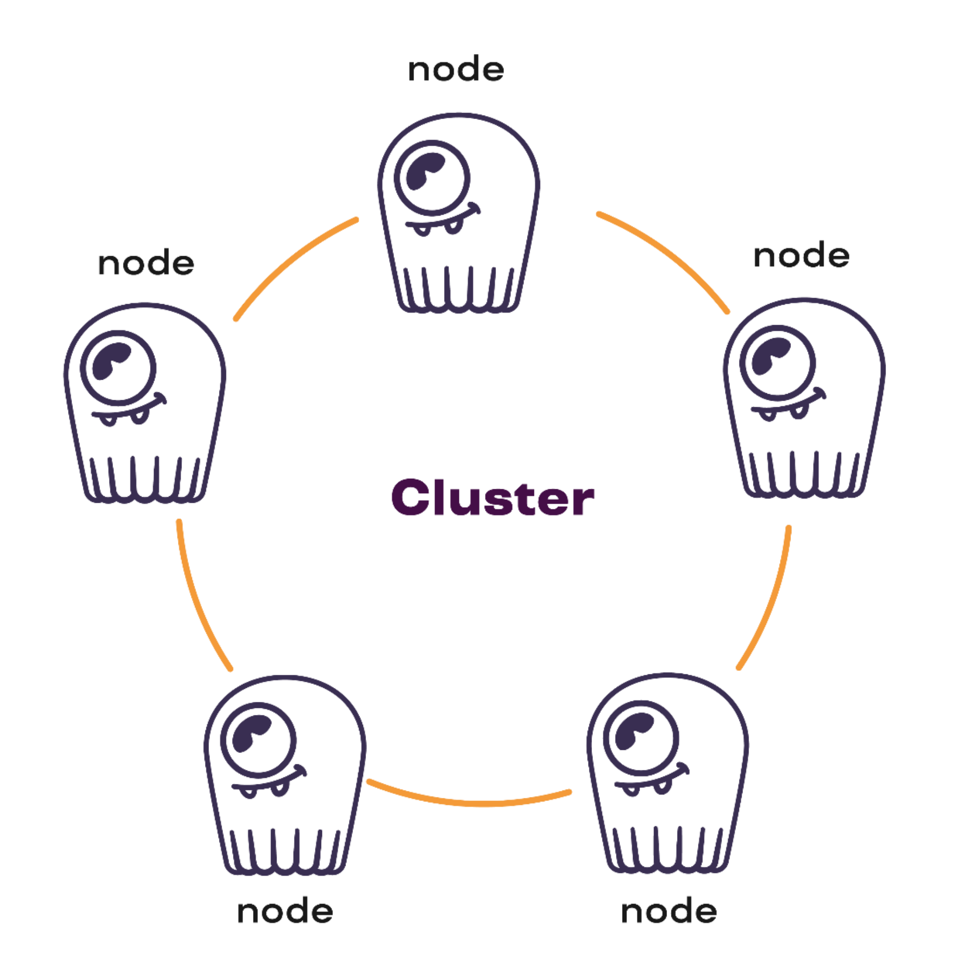
О, Сцилла!

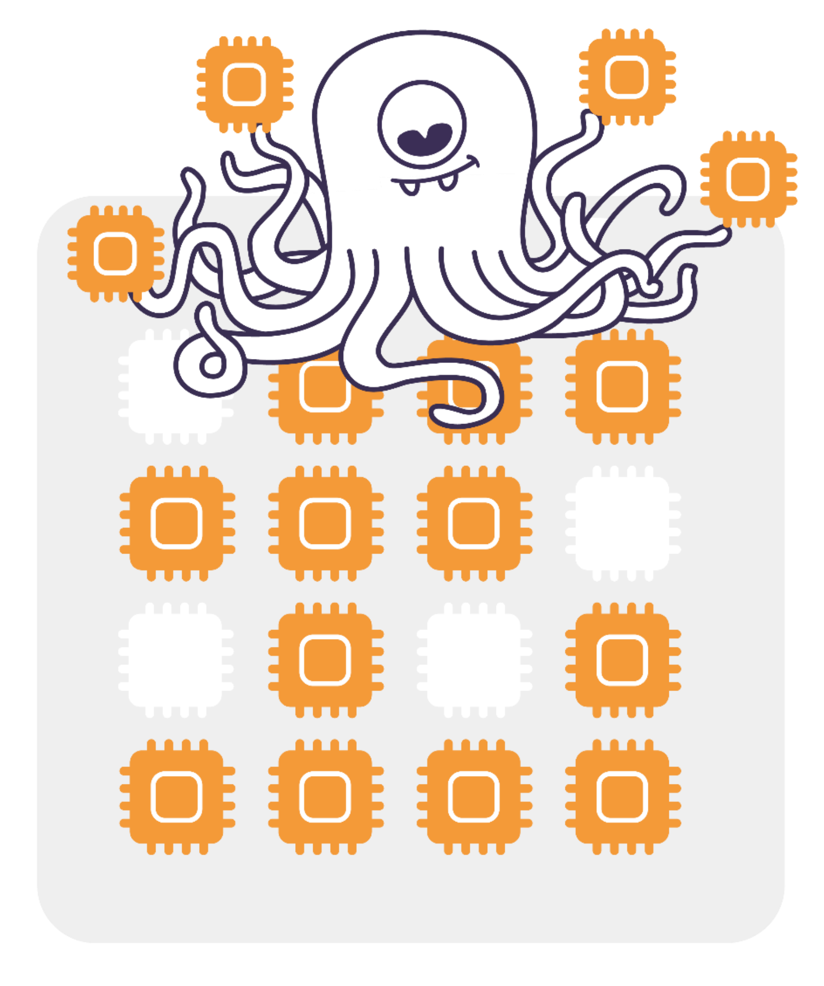


1. Скорость записи — 12500 rps/core (~400 Мбайт/с на 32-ядерном сервере)
2. Реализация на C++
3. Восторженные отзывы (с официального сайта)
4. Эконом-вариант (HDD): 240000 rps (~240 Мбайт/с)
5. Привычный интерфейс (как у CassandraDB)
6. Море доступной документации
7. Открытое сообщество на Github и поддержка в Slack



Её богатый внутренний мир







Инфраструктура

количество серверов: 5

количество CPU на сервере: 32

размер ОП на сервере (Гб): 312

тип жестких дисков: HDD

replication factor: 3

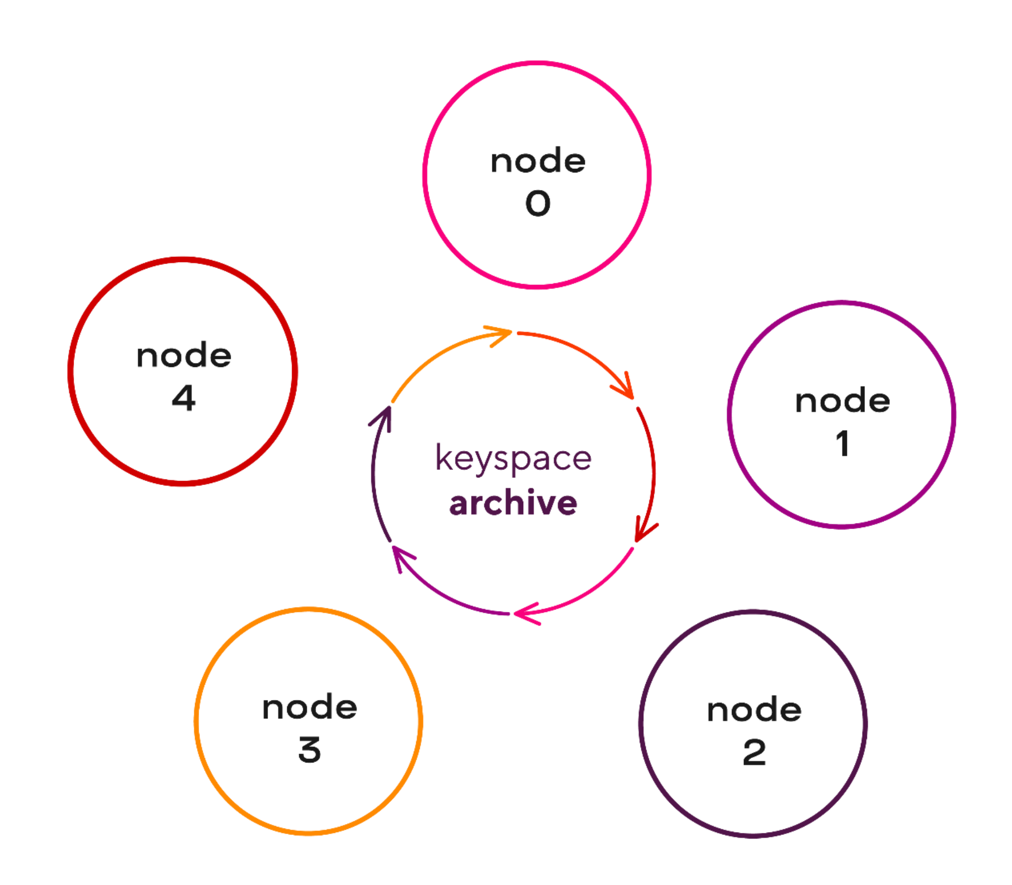




Схема данных

partition key — value

partition — product_id → exec_date | owner | states

row — 50054c8b6f87f912 → 2022-05-31T00:00:00.0 | 12e8ea62818112e8 | "{...}"

⋮

product_id → exec_date | owner | states

50054c8b6f87f902 → 2022-05-30T00:00:00.0 | 12e8ea62818112e8 | "{...}"

table **products**



Этапы архивации

mongoDB

Отбираем записи
для архивации

Этап 1:
копирование

Отправляем
данные в архив

SCYLLA

Отмечаем записи
как отправленные
в архив

Основное хранилище

Архивное хранилище



Этапы архивации

mongoDB

Отбираем
отправленные
в архив записи

Отмечаем записи
как подтверждённые

Этап 2:
валидация

Проверяем
наличие записей
в архиве

Получаем записи,
найденные в архиве

SCYLLA

Основное хранилище

Архивное хранилище



Этапы архивации







Нагрузочное тестирование

cassandra-stress-test: 1 запись — 10 Кбайт

Архивируем по одной записи:

Op rate (скорость выполнения операций): 2431 op/s;

Row rate (скорость добавления записей): 2431 row/s;

Latency mean (средняя задержка операции): 9.8 ms.

Архивируем пачками (по 100 записей за операцию):

Op rate (скорость выполнения операций): 46 op/s;

Row rate (скорость добавления записей): 4634 row/s;

Latency mean (средняя задержка операции): 516.1 ms.

4634 записи в секунду (~45 Мбайт/с) — наш выбор!



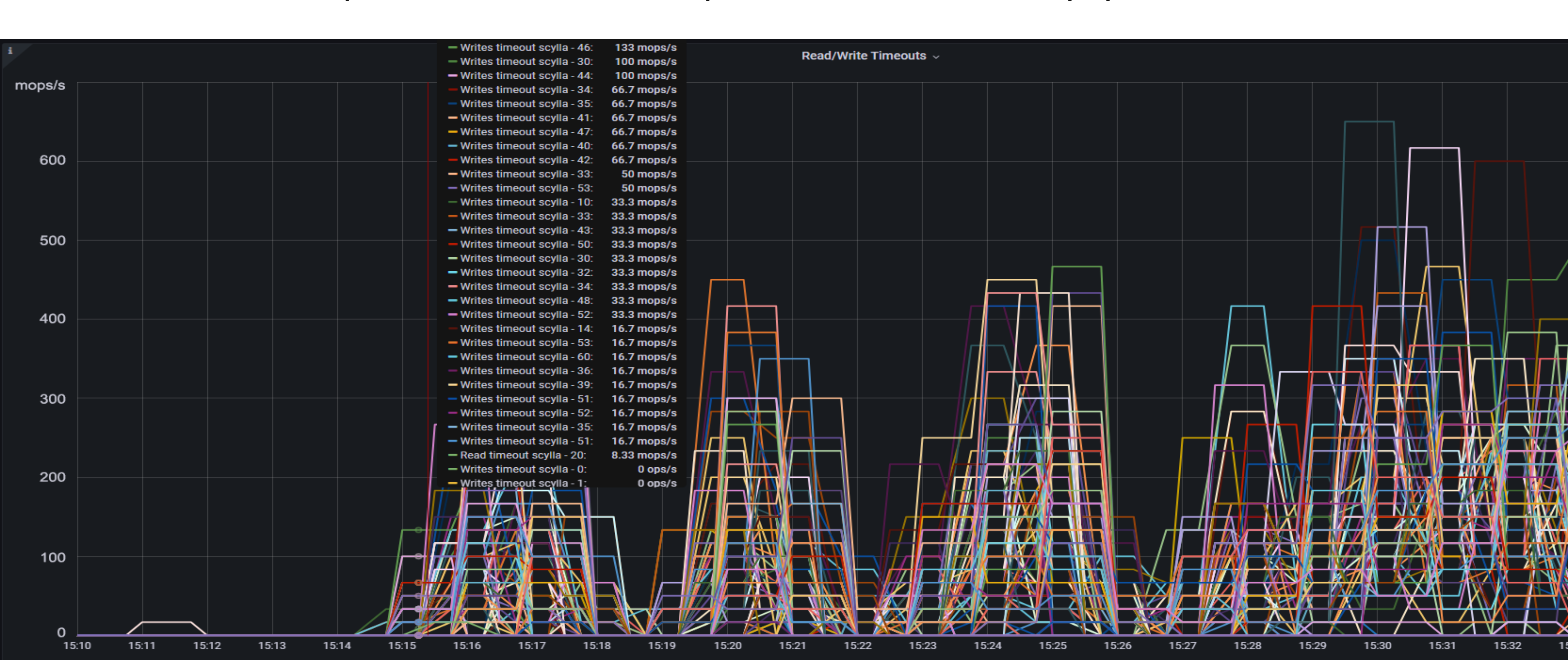
В прод!

	млн записей в сутки	записей в секунду (rps)	Мбайт/с
Требование (от)	**30**	347	3,4
Ожидание	400	4634	45,3
Реальность 1 (копирование)	138	1597	15,6
Реальность 2 (+ валидация и удаление)	**20**	231	2,3
Подождём... посмотрим...	?	?	?



И вот...

- >50% операций записи завершается по тайм-ауту



И вот ещё...

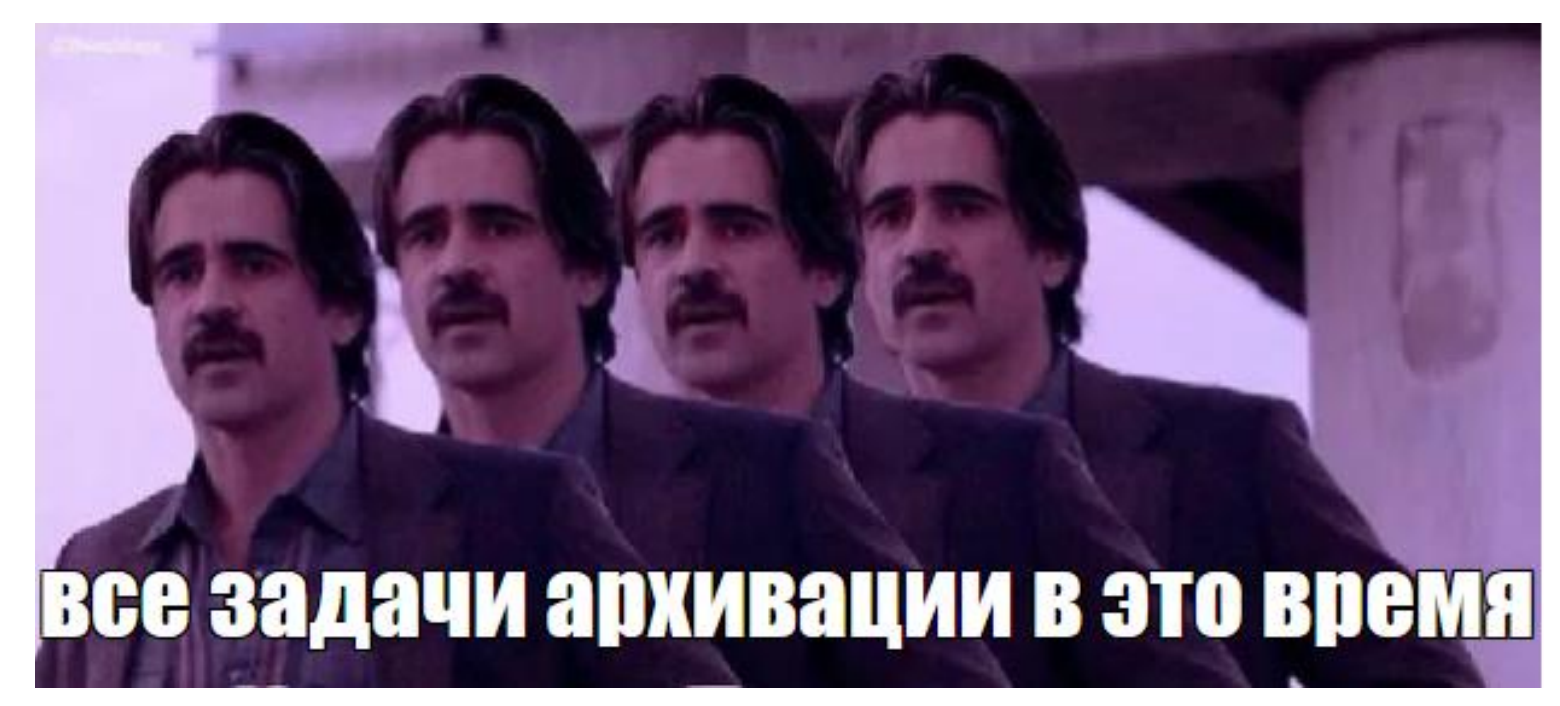
- Падает скорость копирования, валидации (чтения) и архивации в целом (до 3 млн записей в сутки)

	млн записей в сутки	записей в секунду (rps)	Мбайт/с
Требование (от)	**30**	347	3,4
Ожидание	400	4634	45,3
Реальность 3 (дождались)	**3**	35	0,3



И ещё...

- Автокомпактификация занимает до 100% CPU и оказывается неуправляемой





И напоследок...

- Индексы... Какие ещё индексы?

```
large_data - Writing a partition with too many rows
  [products /products_exec_date_idx_index:00000000146428]
  (294334 rows)
```



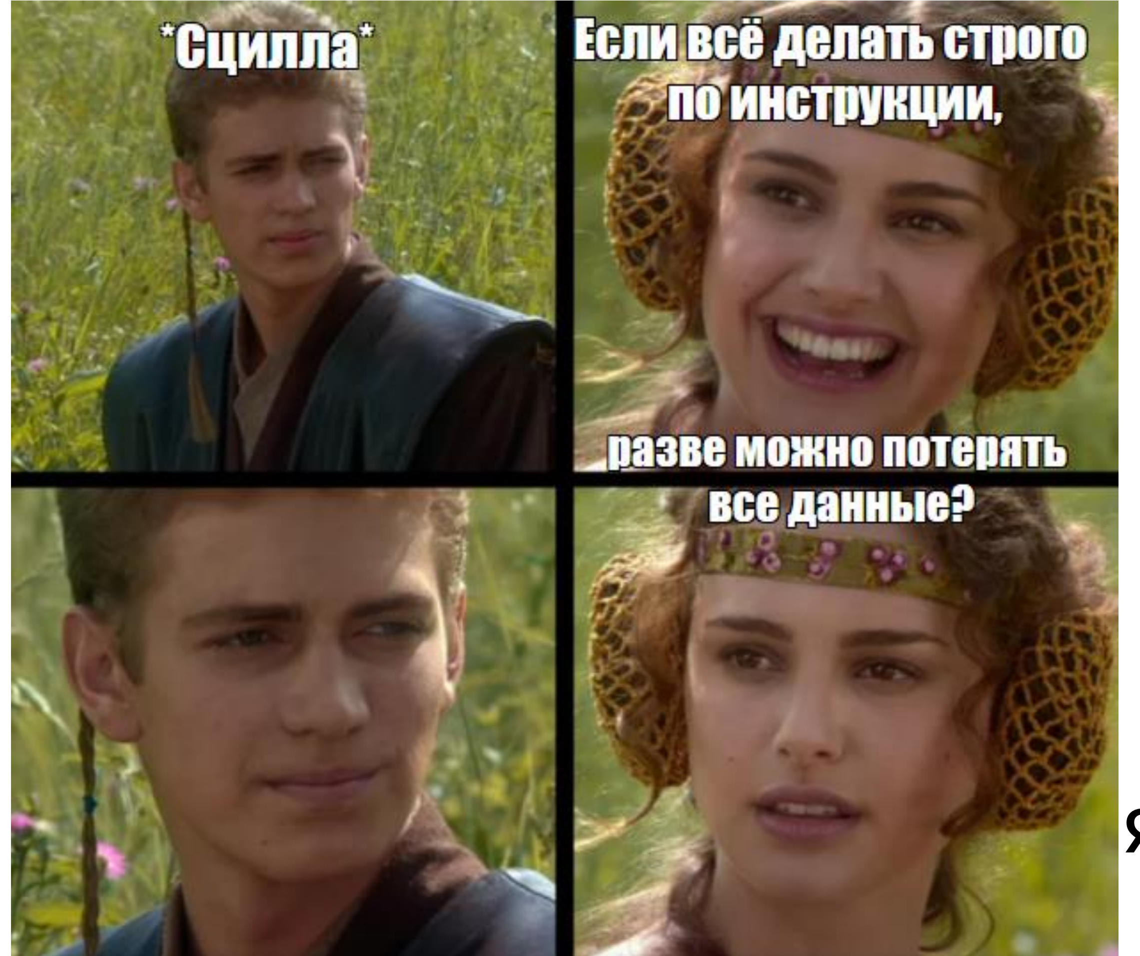
ТУПИК





Что делать?

1. Пробуем переехать на SSD ☹ 👎







Что делать?

1. Пробуем переехать на SSD ☹ 👎
2. Пробуем отключить автокомпактификацию ☹ 👎

nodetool disableautocompaction archive



Что делать?

1. Пробуем переехать на SSD ☹ 👎
2. Пробуем отключить автокомпактификацию ☹ 👎
3. Отказываемся от индексов 🤬 👎

¯_(ツ)_/¯



Что делать?

4. Ослабляем компрессию, переходя с ZSTD на LZ4  

450
400
350
300
250
200
150
100
50
0

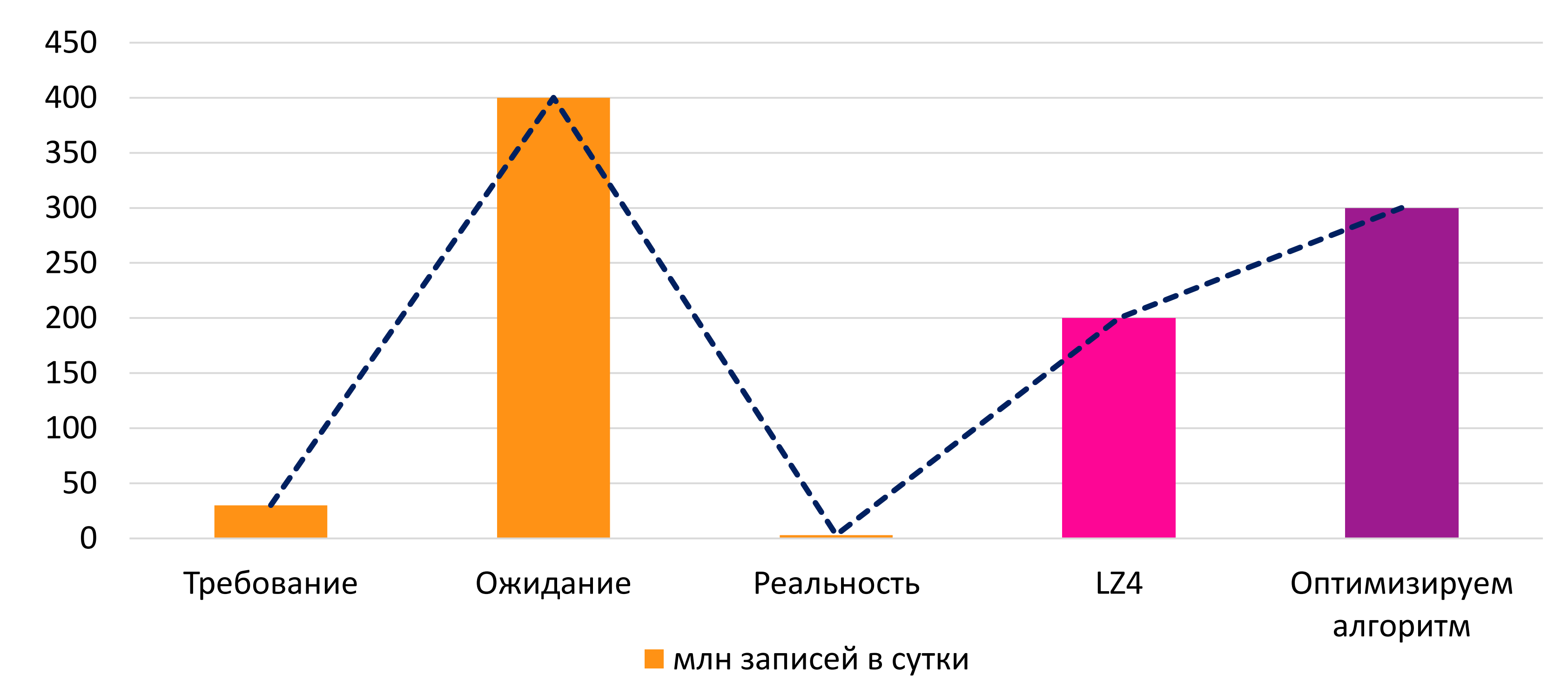
Требование Ожидание Реальность LZ4

млн записей в сутки



Что делать?

4. Ослабляем компрессию, переходя с ZSTD на LZ4  
5. Отказываемся от этапа валидации





Что делать?

4. Ослабляем компрессию, переходя с ZSTD на LZ4 🙂 👍
5. Отказываемся от этапа валидации 🙂 👍
6. Ускоряем работу основного хранилища 🙂 👍





Чего мы добились

	млн записей в сутки	записей в секунду (rps)	Мбайт/с
Требование (от)	**30**	347	3,4
Ожидание	400	4634	45,3
Реальность 1 (копирование)	138	1597	15,6
Реальность 2 (+ валидация и удаление)	20	231	2,3
Реальность 3 (всё плохо)	**3**	35	0,3
Новая реальность	**300**	3472	33,9

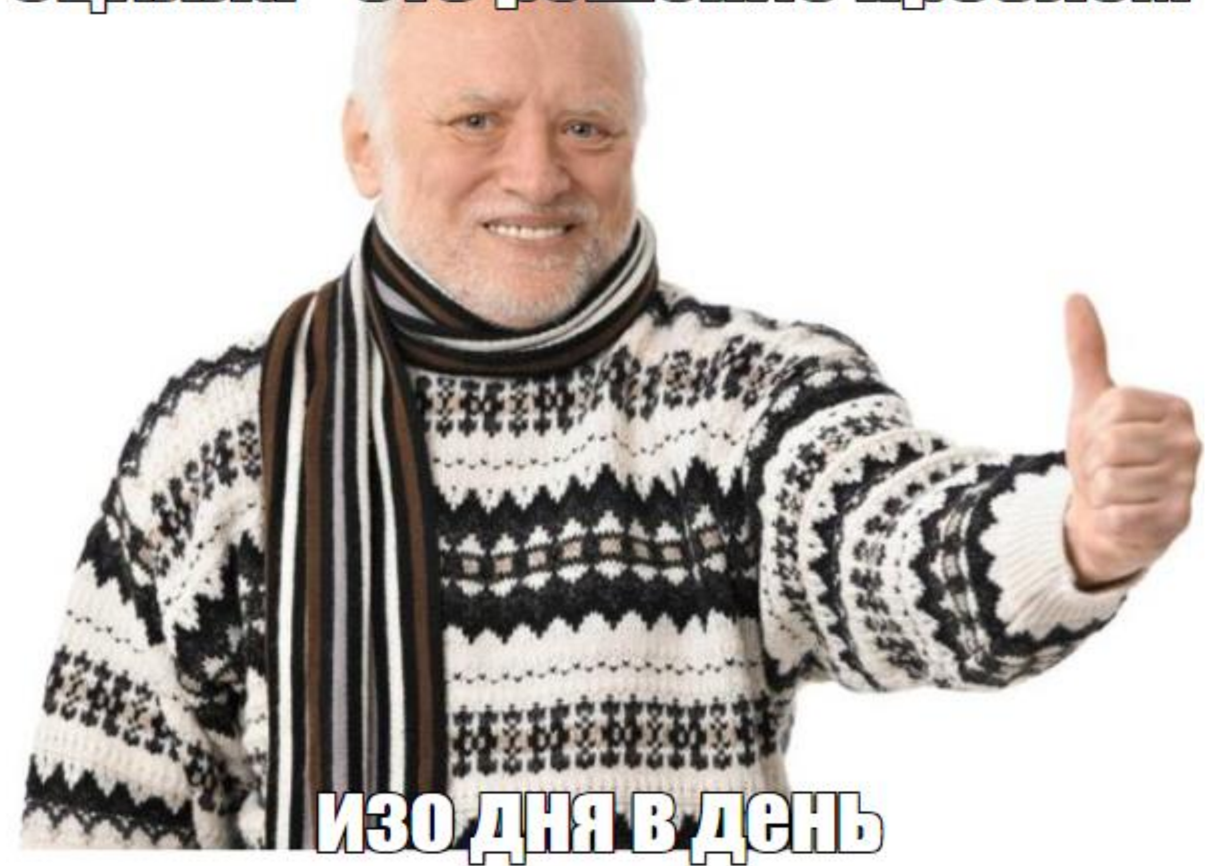


Опыт — сын ошибок трудных

1. Схема данных: key-key-value и никаких индексов
2. Компрессия: Zstd — до первых тайм-аутов, потом — LZ4
3. Компактификация: авторежим со стратегией STCS
4. Валидация: на начальном этапе не помешает
5. Документация: доверяй, но проверяй



И гений — парадоксов друг





Спасибо за внимание!

Оценить доклад



Илья Орлов

e-mail:
ilya.orlov@stm-labs.ru
телеграм-канал Питоняшка:
https://t.me/pythonyashkapy
сайт STM Labs:
https://stm-labs.ru/ru/



Яндекс